MASTER

```
    J U   U PPPP  IIIII TTTTT EEEEE RRRR  RRRR   OOO  M   M       V   V  222          666
    J U   U P   P   I     T   E     R   R R   R O   O MM MM       V   V 2   2        6
    J U   U P   P   I     T   E     R   R R   R O   O M M M       V   V     2       6
    J U   U PPPP    I     T   EEEEE RRRR  RRRR  O   O M M M       V   V   22        6666
    J U   U P       I     T   E     R R   R R   O   O M   M       V   V  2          6   6
J   J U   U P       I     T   E     R  R  R  R  O   O M   M        V V  2     ..    6   6
 JJJ   UUU  P     IIIII   T   EEEEE R   R R   R  OOO  M   M         V   22222 ..     666
```

```
  1    222           1    000          4   4 77777         1    222           1    999          888   222
 11   2   2  :::    11   0   0  :::   4   4     7         11   2   2     /   11   9   9     /  8   8 2   2
  1       2  :::     1   0  00  :::   4   4    7           1       2    /     1   9   9    /   8   8     2
  1     22           1   0 0 0        44444   7            1     22    /      1    9999   /     888    22
  1    2     :::     1   00  0  :::       4  7             1    2     /       1       9  /     8   8  2
  1   2      :::     1   0   0  :::       4  7             1   2     /        1      9  /      8   8 2
11111 22222        11111  000             4  7           11111 22222        11111  999          888  22222
```

possible
improvements
noted

```
MAL/6800 1.3F: 0000
12/18/82 12:03:57; Page 1; Form 1
JUPITERROM26.ASM
                 1: *      V2.6 BOOTROM FOR 6800 CPU CARD WITH IRQ MODIFICATION
                 2: *
                 3: *      BOOT ROM FOR IDB/M6800 RUNNING ON
                 4: *      THE WAVE MATE JUPITER II COMPUTER
                 5: *
                 6: *      WRITTEN BY DENNIS BROWN 3/12/78
                 7: *      COPYRIGHT 1979 WAVE MATE
                 8: *      EDITED 8/25/78 IRA BAXTER FOR IDB VER1.1
                 9: *      EDITED 1/23/79 DENNIS BROWN
                10: *      EDITED 4/4/79 DENNIS PAINTER
                11: *      EDITED 8/30/79 DENNIS PAINTER
                12: *      EDITED 9/4/80 DWP FOR FASTER WM VIDEO
                13: *      EDITED 2/4/81 DWP FOR IDB IN RAM
                14: *      V2.5 11/6/81 DWP FOR DUAL BOOT
                15: *      V2.6 12/18/82 IDB TO MAKE DUAL BOOT WORK
                16: *
                17: *
0000 0000       19:        ABS
 *** Undefined Symbol.
                20: **** EQUATES ****
  FC00          21: TOPAGE  EQU    $FC00   TOP PAGE ON SERIAL SYSTEMS
  0040          22: LINSIZ  EQU    64
  C000          23: SCREEN  EQU    $C000   SCREEN ADDRESS
  0800          24: SCRSIZ  EQU    2048
  0002          25: NULS1   EQU    2
  0002          26: NULS2   EQU    2
  008C          27: SKP2    EQU    $8C     CMPX  #  (SKIP 2 BYTES)
                28: *
                29: ***SYSTEM PAGE***
                30: *
  0080          31:         ORG    $80     RELATIVE TO PAGE
                32: *
0080 0001       33: CURH    RMB    1       CURSOR HIGH HALF
0081 0001       34: CURL    RMB    1       CURSOR LOW HALF
0082 0001       35: CURCHR  RMB    1       CHARACTER CURSOR IS ON TOP OF
0083 0015       36: TABTBL  RMB    21      SYSTEM TAB TABLE FOR MTS
0098 0001       37: CURSTAT RMB    1       CURSOR ON/OFF FLAG
0099 0001       38: TRANSP  RMB    1       TRANSPARENT VIDEO MODE FLAG
                39: *
  00D3          40:         ORG    $D3     RELATIVE FROM SOME PAGE BOUNDRY
                41: *
00D3 0002       42: SYSVARS RMB    2       POINTER TO USER VARIABLES IN MTS
00D5 0003       43: SYSCCI  RMB    3       SYSTEM COMMAND LEVEL VECTOR
00D8 0003       44: SYSRST  RMB    3       SYSTEM SOFT RESET VECTOR
00DB 0003       45: SYSSWI  RMB    3       SYSTEM SOFTWARE INTERRUPT VECTOR
00DE 0002       46: SYSPWR  RMB    2       POWER UP TEST CODE
00E0 0001       47: SYSMSK  RMB    1       SYSTEM INTERRUPT MASK VALUE
00E1 0007       48: SYSCLK  RMB    7       SYSTEM CLOCK
00E8 0003       49: SYSINT7 RMB    3       LEVEL 7 INT FLAG
00EB 0003       50: SYSINT6 RMB    3       LEVEL 6 INT VECTOR
00EE 0003       51: SYSINT5 RMB    3       LEVEL 5 INT VECTOR
00F1 0003       52: SYSINT4 RMB    3       LEVEL 4 INT VECTOR
00F4 0003       53: SYSINT3 RMB    3       LEVEL 3 INT VECTOR
00F7 0003       54: SYSINT2 RMB    3       LEVEL 2 INT VECTOR
00FA 0003       55: SYSINT1 RMB    3       LEVEL 1 INT VECTOR
```

Add "Wire" mode to ROM.

Mod ROM so IDB can use top $80 bytes, ROM uses $70-$7F.

These are deletable.

Many of these are deleteable.

Simplify code that refers to this stuff.

```
00FD 0003        56: SYSINTO RMB     3       LEVEL 0 INT VECTOR
  00FE           57:         ORG     $FE     SYSTEM PAGE POINTER          ← delete this ugly thing.
00FE 0001        58: SYSPG   RMB     1       POINTER TO SYSTEM PAGE
00FF 0001        59: RUNNINGENCRYPTED        RMB     1        FLAG, IN ENCRYPTED MODE    ← Move to
                                                                                           system page.
```

```
MAL/6800 1.3F: 00FF
12/18/82 12:03:57; Page 3; Form 2
JUPITERROM26.ASM
                 1: *
0100 0000        2:         ABS
 *** Undefined Symbol.
                 3: *       START OF THE ROM
                 4:
                 5: *
  FC00           6:         ORG     $FC00   BOOT ROM STARTING LOCATION
                 7: ***VECTOR JUMP TABLE***
                 8: *
FC00 7EFD25      9: BOOTV   JMP     BOOT    BOOT LOAD VECTOR
FC03 7EFE11     10: INICV   JMP     INIS1   INIT BOTH SERIAL & VIDEO
                11: *
FC06 7EFDE6     12: PUTCV   JMP     PUTS1   CONSOLE OUTPUT
FC09 7EFDC7     13: GETCV   JMP     GETS1   CONSOLE INPUT
FC0C 7EFDBD     14: TSTCV   JMP     TSTS1   CONSOLE INPUT TEST
FC0F 7EFC42     15: ESCCV   JMP     ESCC    CONSOLE INPUT ESCAPE TEST
                16: *
FC12 7EFDB1     17: INIFV   JMP     INIS2   INIT FILE DEVICE
FC15 7EFE0A     18: PUTFV   JMP     PUTS2   FILE OUTPUT
FC18 7EFDD5     19: GETFV   JMP     GETS2   FILE INPUT
FC1B 7EFE02     20: OONFV   JMP     OUTFON  FILE OUTPUT START
FC1E 7EFE05     21: OOFFFV  JMP     OUTFOFF FILE OUTPUT STOP
FC21 7EFDFC     22: IONFV   JMP     INFON   FILE INPUT START
FC24 7EFDFF     23: IOFFFV  JMP     INFOFF  FILE INPUT STOP
                24: *
FC27 7EFDB1     25: INILV   JMP     INIS2   INIT LIST DEVICE
FC2A 7EFE0A     26: PUTLV   JMP     PUTS2   LIST OUTPUT
FC2D 7EFD94     27: OONLV   JMP     RETURN  LIST OUTPUT START
FC30 7EFD94     28: OOFFLV  JMP     RETURN  LIST OUTPUT STOP
                29: *
FC33 01         30: INTDISV NOP             DISABLE INTERRUPTS VECTOR
FC34 0F         31:         SEI
FC35 39         32:         RTS
FC36 0E         33: INTENV  CLI             ENABLE INTERRUPTS VECTOR
FC37 39         34:         RTS
FC38 01         35:         NOP             MAINTAIN OLD 3 BYTE JMP SPACE
FC39 3B         36: NOINT   RTI             RETURN FROM INTERRUPT VECTOR
FC3A 7EFE16     37: TSTMEM  JMP     TESTRAM
                38: *
FC3D 8DF4       39: IDBNMI  BSR     INTDISV IDB NMI ENTRY POINT, STOP INTS    these could be wasted.
FC3F 7EF008     40:         JMP     $F008   ENTER THE DEBUGGER
                41: *
FC42 8DC8       42: ESCC    BSR     TSTCV   CONSOLE ESCAPE INPUT TEST RETURNS
FC44 2606       43:         BNE     ESCCE   ZERO FLAG SET IF ESCAPE HIT
FC46 8DC1       44:         BSR     GETCV   GET DATA
FC48 847F       45:         ANDA    #$7F    MASK TO 7 BITS
FC4A 811B       46:         CMPA    #$1B    CK IF ESCAPE CHAR
FC4C 39         47: ESCCE   RTS
                48: *
```

```
                   1: *
                   2: *       SYSTEM PAGE INITIAL VALUES
                   3: *
  FC4D             4: SYSINIT EQU      *
FC4D 7EFC3D        5:         JMP      IDBNMI  SYSCCI
FC50 7EF005        6:         JMP      $F005   SYSRST
FC53 7EFC3D        7:         JMP      IDBNMI  SYSSWI
FC56 A55A          8:         FDB      $A55A   SYSPWR
FC58 00            9:         FCB      0       SYSMSK
FC59 00000000     10:         FCB      0,0,0,0,0,0,0   SYSCLK
FC60 020000       11:         FCB      2,0,0   SYSINT7 FLAG
FC63 7EFC39       12:         JMP      NOINT   SYSINT6
FC66 7EFC39       13:         JMP      NOINT   SYSINT5
FC69 7EFC39       14:         JMP      NOINT   SYSINT4
FC6C 7EFC39       15:         JMP      NOINT   SYSINT3
FC6F 7EFC39       16:         JMP      NOINT   SYSINT2
FC72 7EFC39       17:         JMP      NOINT   SYSINT1
FC75 7EFC39       18:         JMP      NOINT   SYSINT0
                  19: *
                  20: *       I/O INTERRUPT ENTRY POINTS
                  21: *
                  22: *       ALL INTS VECTOR THROUGH SYSTEM PAGE
                  23: *
  FC78            24: LEVEL0  EQU      *       INTERRUPT LEVEL 0
FC78 86FD         25:         LDAA     #SYSINT0&$FF
FC7A 36           26:         PSHA
FC7B 96FE         27:         LDAA     SYSPG
FC7D 36           28:         PSHA
FC7E 39           29:         RTS
                  30: *       LDX      SYSPG
                  31: *       JMP      SYSINT0,X
                  32: *
  FC7F            33: LEVEL7  EQU      *       INTERRUPT LEVEL 7 (FPI)
                  34: *       LDAA     RUNNINGENCRYPTED
FC7F B6           35:         FCB      $B6     LDAA
FC80 00FF         36:         FDB      RUNNINGENCRYPTED
FC82 26B5         37:         BNE      NOINT
FC84 DEFE         38:         LDX      SYSPG
FC86 6FE8         39:         CLR      SYSINT7,X       SET FPI FLAG
FC88 CEC350       40:         LDX      #50000  SET TIMER
FC8B 09           41: FPITIM  DEX
FC8C 26FD         42:         BNE      FPITIM
                  43: *
  FC8E            44: FINT2   EQU      *       FPI END
FC8E DEFE         45:         LDX      SYSPG
FC90 6CE8         46:         INC      SYSINT7,X       CHANGE FPI FLAG BACK TO NORMAL
FC92 6ED5         47:         JMP      SYSCCI,X        JUMP TO COMMAND LEVEL
                  48: *
```

simplify

simplify — also speeds up computer!!

```
                  *
                  *       SYSTEM RESET ENTRY POINT
                  *
 FC94             HARDRESET       EQU     *
FC94 7EFC9F               JMP     RESET9
FC97 02000000             FCB     2,0,0,0,0,0,0,0 SERIAL NUMBERS
FC9F 8E00FF       RESET9  LDS     #$FF    PUSH RET ADDR INTO SYSPG PTR
                  *       LDAA    RUNNINGENCRYPTED
FCA2 B6                   FCB     $B6     LDAA
FCA3 00FF                 FDB     RUNNINGENCRYPTED
FCA5 2710                 BEQ     NOTENCRYPT
FCA7 8600                 LDAA    #0
FCA9 BDFC4C       CLR     JSR     ESCCE
FCAC CEBFFF               LDX     #$BFFF
 FCAF             LOOP    SET     *
FCAF 6F00                 CLR     0,X
FCB1 09                   DEX
FCB2 26FB                 BNE     LOOP
FCB4 4A                   DECA
FCB5 2AF2                 BPL     CLR
 FCB7             NOTENCRYPT      EQU     *
FCB7 CEFC00               LDX     #TOPAGE
FCBA DFFE                 STX     SYSPG
                  *
FCBC 7A00FE       TOPLOOP DEC     SYSPG   TRY NEXT PAGE
FCBF DEFE                 LDX     SYSPG
FCC1 A600                 LDAA    0,X     WILL THE LOCATION HOLD
FCC3 43                   COMA            IT'S COMPLEMENT?
FCC4 A700                 STAA    0,X
FCC6 A000                 SUBA    0,X
FCC8 26F2                 BNE     TOPLOOP B/ NO MEMORY HERE
FCCA 6300                 COM     0,X     RESTORE THE BYTE
FCCC 6DE8                 TST     SYSINT7,X       FPI SWITCH DOWN?
FCCE 2707                 BEQ     RESET   B/ YES HIT FULL RESET
                  *
FCD0 EEDE                 LDX     SYSPWR,X        GET POWER UP CODE
FCD2 8CA55A               CPX     #$A55A
FCD5 272D                 BEQ     SOFTRESET       POWER UP RESET IF CODE TRASHED.
                  *
 FCD7             RESET   EQU     *       POWER UP RESET ENTRY CODE
FCD7 DEFE                 LDX     SYSPG   RESTORE SYSTEM PAGE
FCD9 8EFC4C               LDS     #SYSINIT-1      INITIALIZE SYSTEM PAGE
                  *
                  *
FCDC C6D5                 LDAB    #$D5
                  *
 FCDE             SYSLOOP EQU     *       COPY LOOP
FCDE 32                   PULA
FCDF A7D5                 STAA    $D5,X
FCE1 08                   INX
FCE2 5C                   INCB
FCE3 26F9                 BNE     SYSLOOP
                  *
FCE5 FEF001               LDX     $F001   CHECK TO SEE IF IDB IS THERE
FCE8 8C01BD               CPX     #$01BD
```

simplify

```
FCEB 2607                 BNE     SETBOOT B/ NOT HERE
FCED B6F043               LDAA    $F043   GET IOB SCRATCH PAGE
FCF0 91FE                 CMPA    $FE    SEE IF SYSTEM HAS RAM THERE
FCF2 2310                 BLS     SOFTRESET       B/ YES, DON'T BOOT
FCF4 DEFE        SETBOOT  LDX     SYSPG
FCF6 86FC                 LDAA    #BOOTV/256      IF NOT MAKE RESET LEVEL
FCF8 C600                 LDAB    #BOOTV&255
FCFA A709                 STAA    SYSRST+1,X      POINT TO BOOT ROUTINE
FCFC E70A                 STAB    SYSRST+2,X
FCFE C639                 LDAB    #NOINT&255      NO IOB HERE SO POINT FPI
FD00 A7D6                 STAA    SYSCCI+1,X      VECTOR TO NOINT SUBR
FD02 E707                 STAB    SYSCCI+2,X
                 *
                 *        SOFT RESET ENTRY POINT
                 *
  FD04           SOFTRESET        EQU     *
FD04 9EFE                 LDS     SYSPG   INIT STACK POINTER
FD06 31                   INS
FD07 31                   INS
FD08 31                   INS             STAY OFF RAM PAGE BELOW SYSPAGE
FD09 BDFC03               JSR     INICV   INIT CONSOLE DEVICE
FD0C BDFC12               JSR     INIFV   INIT DEFAULT DEVICE
FD0F BDFC27               JSR     INILV   INIT LIST DEVICE
FD12 9EFE                 LDS     SYSPG   NOW SET UP CONTEXT BLOCK
FD14 30                   TSX             POINT TO SYSPG (PLUS 1)
FD15 8625                 LDAA    #BOOT&255       SET UP USER ENVIRONMENT
FD17 A706                 STAA    6,X     SO THAT A GO COMMAND
FD19 86FD                 LDAA    #BOOT/256       WILL BOOTSTRAP IN A BIGGER
FD1B A705                 STAA    5,X     SYSTEM PROGRAM.
FD1D A700                 STAA    0,X     USE THE BOOT LOADER PAGE $FD TO SET CCR
                 *                AS INTERUPTS OFF
FD1F 6ED7                 JMP     SYSRST-1,X      AND GO TO COMMAND LEVEL
                 *
                 *        SWI ENTRY POINT, VECTORS THROUGH THE SYSTEM PAGE SYSSWI
                 *
  FD21           SWIV     EQU     *
FD21 DEFE                 LDX     SYSPG
FD23 6EDB                 JMP     SYSSWI,X        JUMP THROUGH VECTOR
```

```
             *       DUAL BOOT LOADER FOR EITHER FDI-125 & PERSCI 277 OR FDI-127 & T&E
             *       BOOT TRIES TO LOAD FIRST SECTOR FROM EITHER DEVICE WHICH INDICATES
             *       READY STATUS.  IF A SECTOR ZERO IS NOT FOUND IT TRIES TO FIND A SECTOR
             *       NUMBER ONE.  IF SECTOR NUMBER ONE IS FOUND ON AN 8 INCH DRIVE IT
             *       ASSUMES IBM FORMAT DISK AND COMPLEMENTS THE DATA.  IF THE FIRST
             *       INSTRUCTION IS NOT A LOAD STACK IMMEDIATE ($8E) IT DOES NOT EXECUTE
             *       THE LOADED BOOTSTRAP.
FD25 8D3A    BOOT    BSR     INITIALIZE      INITIALIZE WHATEVER HARDWARE IS ON SYSTEM
FD27 8199            CMPA    #$99    ANY ERRORS?
FD29 26FA            BNE     BOOT    B/ WELL TRY AGAIN

FD2B 4F      RETRY   CLRA            SET SECTOR 0
FD2C 8D22            BSR     READSECT        TRY TO READ IT
FD2E 2716            BEQ     TSTBOOT B/ NO ERRORS CK FOR GOOD BOOTSTRAP DATA
FD30 818D            CMPA    #$8D    RECORD NOT FOUND?
FD32 26F7            BNE     RETRY   B/ NO, TRY AGAIN
FD34 8601            LDAA    #1      TRY SECTOR ONE
FD36 8D18            BSR     READSECT
FD38 26F1            BNE     RETRY   B/ ERROR, TRY SECTOR 0 AGAIN
FD3A 8D71            BSR     ISMINI  READ MINI SECTOR 1?
FD3C 2708            BEQ     TSTBOOT B/ YES, DON'T COMPLEMENT DATA
FD3E CE0080          LDX     #$17F-$FF       MUST BE IBM FORMAT, COMPLEMENT IT
FD41 63FF    COMDATA COM     $FF,X
FD43 09              DEX
FD44 26FB            BNE     COMDATA

FD46 B60100  TSTBOOT LDAA    $100    FIRST OPCODE MUST BE
FD49 818E            CMPA    #$8E    LOAD STACK IMMEDIATE
FD4B 26D8            BNE     BOOT    OR WE ABORT THE WHOLE SHOW
FD4D 7E0100          JMP     $100    GO EXECUTE THE BOOTSTRAP

             * READ SELECTED SECTOR SUBROUTINE

FD50 8D5B    READSECT        BSR     ISMINI  CK WHICH DRIVE IS SELECTED
FD52 2701            BEQ     SETSECT B/ MINI, DON'T COMPLEMENT SECTOR NUMBER
FD54 43              COMA
FD55 A706    SETSECT STAA    6,X     SELECT 17XX SECTOR
FD57 8601            LDAA    #1      SET READ PAGE
FD59 A702            STAA    2,X
FD5B 17              TBA             GET READ COMMAND INTO ACCA
FD5C 8D27            BSR     CMD     ISSUE READ COMMAND
FD5E 819D            CMPA    #$9D    CK FOR ERRORS
FD60 39              RTS             AND RETURN THEM

             * INITIALIZE HARDWARE

 FD61        INITIALIZE      EQU     *
FD61 CEFFA0  INI8IN  LDX     #$FFA0  SET BASE ADDRESS OF PERSCI
FD64 8D2F            BSR     INIHWD
FD66 A103            CMPA    3,X     DOES THE HARDWARE EXIST?
FD68 260A            BNE     INI5IN  B/ NO, TRY THE MINI-FLOPPY
FD6A A604            LDAA    4,X     IS DRIVE READY?
FD6C 2A06            BPL     INI5IN  B/ NO, TRY 5" DRIVE
FD6E 86F0            LDAA    #$F0    ISSUE RESTORE COMMAND
FD70 C673            LDAB    #$73    SET UP READ COMMAND FOR MAIN
```

```
FD72 2011       56:         BRA     CMD
                57:
FD74 CEFF80     58: INI5IN  LDX     #$FF80  SET BASE ADDRESS OF T&E
FD77 8D1C       59:         BSR     INIHWD
FD79 A103       60:         CMPA    3,X     DOES THE HARDWARE EXIST?
FD7B 26E4       61:         BNE     INI8IN  B/ NO, TRY THE 8 INCH DRIVE
FD7D A604       62:         LDAA    4,X     IS DRIVE READY?
FD7F 2BE0       63:         BMI     INI8IN  B/ NO, TRY 8" DRIVE
FD81 860D       64:         LDAA    #$0D    SET RESTORE FLAG
FD83 C680       65:         LDAB    #$80    SET UP READ COMMAND FOR MAIN
                66:
FD85 A704       67: CMD     STAA    4,X     ISSUE COMMAND TO HARDWARE
FD87 A601       68: FDWAIT  LDAA    1,X     WAIT FOR DONE
FD89 2AFC       69:         BPL     FDWAIT
FD8B A604       70:         LDAA    4,X     GET CONTROLLER STATUS
FD8D 8D1C       71:         BSR     DISKACKDONE     ACK DONE FLAGS, WHICH DISK DRIVE ?
FD8F 2601       72:         BNE     FDCMDX  B/ GOT 8 INCH STATUS, EXIT
FD91 43         73:         COMA            CONVERT 5 INCH STATUS TO LOOK LIKE 8 INCH STATUS
FD92 849D       74: FDCMDX  ANDA    #$9D    STRIP GARBAGE BITS
FD94 39         75: RETURN  RTS
                76:
                77:
                78: * INITIALIZE HARDWARE POINTED TO BY X-REG
                79:
FD95 4F         80: INIHWD  CLRA            GET A ZERO
FD96 A700       81:         STAA    0,X     INITIALIZE PIA
FD98 A701       82:         STAA    1,X
FD9A 4A         83:         DECA            GET A $FF
FD9B A702       84:         STAA    2,X
FD9D A703       85:         STAA    3,X
FD9F 862C       86:         LDAA    #$2C
FDA1 A700       87:         STAA    0,X
FDA3 8616       88:         LDAA    #$16
FDA5 A701       89:         STAA    1,X
FDA7 8640       90:         LDAA    #$40    (60 FOR MINI, 48 FOR 8IN, HOPE THIS WORKS)
FDA9 A703       91:         STAA    3,X     SELECT UNIT 0
  FDAB          92: DISKACKDONE ; ACKNOWLEDGE DISK DONE FLAGS
FDAB A503       93:         BITA    3,X     CLR DONE FLAGS
  FDAD          94: ISMINI ; CHECK FOR MINI FLOPPY: RETURN Z SET IF MINI
FDAD 8CFF80     95:         CPX     #$FF80  USING MINIFLOPPY HARDWARE BASE ?
FDB0 39         96:         RTS
```

```
                  1: *
                  2: *       SERIAL 2 INITIALIZATION
                  3: *
                  4: *       M6850 ACIA BASED AT $FFC4
                  5: *       INIT FOR 1 START/ 8 DATA/ 2 STOP BITS
                  6: *       /16 CLOCK/ NO INTERRUPTS ENABLED
                  7: *
FDB1 CEFFC4       8: INIS2   LDX     #$FFC4
FDB4 8603         9: INISA   LDAA    #3      GET RESET CODE
FDB6 A700        10:         STAA    0,X     AND STUFF IT INTO ACIA
FDB8 8611        11:         LDAA    #$11    GET INIT CODE
FDBA A700        12:         STAA    0,X     AND STUFF IT
FDBC 39          13:         RTS             EXIT
                 14: *
                 15: *       SERIAL 1 INPUT DONE TEST
                 16: *
                 17: *       M6850 ACIA BASED AT $FFC0
                 18: *       SETS STATUS ZERO IF DONE
                 19: *
FDBD CEFFC0      20: TSTS1   LDX     #$FFC0
FDC0 E600        21: TSTSA   LDAB    0,X     GET STATUS BITS
FDC2 C401        22:         ANDB    #1
FDC4 C801        23:         EORB    #1      TEST BOTTOM BIT
FDC6 39          24:         RTS             EXIT
                 25: *
                 26: *       SERIAL 1 INPUT
                 27: *
                 28: *       M6850 ACIA BASED AT $FFC0
                 29: *       RETURNS CHARACTER IN A
                 30: *
FDC7 CEFFC0      31: GETS1   LDX     #$FFC0
FDCA 8DF4        32: GETSA   BSR     TSTSA
FDCC 26FC        33:         BNE     GETSA   WAIT FOR INPUT DONE
FDCE A601        34:         LDAA    1,X     GET CHARACTER
FDD0 847F        35:         ANDA    #$7F    MASK IT
FDD2 27F6        36:         BEQ     GETSA   IGNORE NULLS
FDD4 39          37:         RTS             EXIT
                 38: *
                 39: *       SERIAL 2 INPUT
                 40: *
FDD5 CEFFC4      41: GETS2   LDX     #$FFC4
FDD8 20F0        42:         BRA     GETSA
                 43:
                 44: *
                 45: *       SERIAL 1 OUTPUT TEST
                 46: *
                 47: *       M6850 ACIA BASED AT $FFC0
                 48: *       SETS STATUS ZERO IF OUTPUT DONE
                 49: *
  FDDA           50: TSTS1O  EQU     *
FDDA CEFFC0      51:         LDX     #$FFC0
FDDD 37          52: TSTSAO  PSHB            SAVE USR ACCB
FDDE E600        53:         LDAB    0,X
FDE0 C402        54:         ANDB    #$2
FDE2 C802        55:         EORB    #$2     TEST BIT 1 OF ACIA
```

```
FDE4 33                 PULB            RESTORE USR ACCB
FDE5 39                 RTS             EXIT
                *
                *       SERIAL 1 OUTPUT
                *
                *       CHARACTER TO OUTPUT IS IN A
                *       A LINE FEED GETS SOME NULLS AFTER IT
                *
  FDE6          PUTS1   EQU     *
FDE6 C602               LDAB    #NULS1  SET NUMBER OF NULLS AFTER A LF
FDE8 CEFFC0             LDX     #$FFC0
  FDEB          PUTSA   EQU     *
FDEB 8DF0       PUTSB   BSR     TSTSAO
FDED 26FC               BNE     PUTSB   WAIT FOR OUTPUT DONE
FDEF A701               STAA    1,X     STUFF CHR INTO ACIA
FDF1 810A               CMPA    #$A
FDF3 2606               BNE     PUTS1E  EXIT IF NOT A LF
FDF5 4F         PUTSN   CLRA            PUT A NULL
FDF6 8DF3               BSR     PUTSA   PUT A NULL
FDF8 5A                 DECB
FDF9 26FA               BNE     PUTSN
FDFB 39         PUTS1E  RTS             EXIT

                *       FILE READER START --OUTPUT DC1
                *       FILE READER STOP ---OUTPUT DC3
                *       FILE PUNCH  START --OUTPUT DC2
                *       FILE PUNCH  STOP ---OUTPUT DC4
                *
FDFC 8611       INFON   LDAA    #$11    INPUT ON
FDFE 8C                 SKP2
FDFF 8613       INFOFF  LDAA    #$13    INPUT OFF
FE01 8C                 SKP2
FE02 8612       OUTFON  LDAA    #$12    OUTPUT ON
FE04 8C                 SKP2
FE05 8614       OUTFOFF LDAA    #$14    OUTPUT OFF
FE07 7EFC15             JMP     PUTFV   OUTPUT CODE TO FILE DEVICE

                *
                *       SERIAL 2 OUTPUT
                *
                *       ACIA BASED AT $FFC4
                *
FE0A C602       PUTS2   LDAB    #NULS2
FE0C CEFFC4             LDX     #$FFC4
FE0F 20DA               BRA     PUTSA
                *
                *       SERIAL 1 INITIALIZATION
                *
FE11 CEFFC0     INIS1   LDX     #$FFC0
FE14 209E               BRA     INISA

```

```
 0000         UPPER   EQU     0       TWO BYTES FOR UPPER LIMIT
 0002         LOWER   EQU     2       TWO BYTES FOR LOWER LIMIT
 0004         COUNT   EQU     4       ONE BYTE
 0005         PASS    EQU     5       ONE BYTE
 0006         PATT    EQU     6       ONE BYTE
              * MEMORY TEST
              *       DOES PATTERN TEST ON ALL OF MEMORY
              *       DISPLAYS SLOWLY INCREMENTING PATTERN NUMBER 0 TO $FF
              *       PRINTS "VERIFIED" OR "ERR @XXXX GOOD=GG BAD=BB"

FE16 CE0000   TESTRAM LDX     #0
FE19 DF01             STX     UPPER+1 SET LOW BYTE OF UPPER & HI BYTE OF LOWER
FE1B DF04             STX     COUNT   SET COUNT & PASS
FE1D DF06             STX     PATT    SET PATT
FE1F 8607             LDAA    #PATT+1 SET LOWER LIMIT
FE21 9703             STAA    LOWER+1
FE23 96FE             LDAA    SYSPG   SET UPPER LIMIT
FE25 4C               INCA
FE26 9700             STAA    UPPER
FE28 0F               SEI

              * DO PATTERN SENSITIVITY TEST

FE29 730005   PTEST1  COM     PASS
FE2C 261D             BNE     INIT
FE2E 9606             LDAA    PATT
FE30 4C               INCA
FE31 840F             ANDA    #$F
FE33 9706             STAA    PATT
FE35 2614             BNE     INIT
FE37 DE02             LDX     LOWER   LONG TST COMES HERE FIRST
FE39 08               INX             SO LOCATION LOWER+1 IS NOT TESTED
FE3A DF02             STX     LOWER
FE3C D604             LDAB    COUNT
FE3E 5C               INCB
FE3F D704             STAB    COUNT   SAVE NEW COUNT
FE41 2759             BEQ     DONE    MEMORY DIAGNOSTIC DONE
FE43 BDFEC4           JSR     PRINTSTR
FE46 0888             FCB     $8,$8+$80
FE48 17               TBA             SET COUNT IN ACCA
FE49 8D5E             BSR     PUTHEX  DISPLAY

              * FILL MEMORY FROM LOWER TO UPPER WITH PATTERN OF ALTERNATING
              * BITS WHICH SHIFT FROM $55 TO $AA ON BOUNDARIES DEFINED BY
              * THE VARIABLE PATT.  IF PATT =  0 THEN FILL ALL MEMORY WITH
              * AN ALTERNATING PATTERN.

FE4B DE02     INIT    LDX     LOWER   FILL FROM LOWER TO UPPER
FE4D 8655             LDAA    #$55
FE4F 9805             EORA    PASS
FE51 D606     INIT1   LDAB    PATT
FE53 43               COMA
FE54 A700     INIT2   STAA    X
FE56 08               INX
FE57 9C00             CPX     UPPER
```

```
FE59 2708                BEQ    CHKUP
FE5B 5D                  TSTB             IF PATT = 0
FE5C 27F6                BEQ    INIT2     THEN FILL ALL MEMORY W/ PASS=[0 OR FF]
FE5E 5A                  DECB             ELSE ALTERNATE BIT PATTERN
FE5F 26F3                BNE    INIT2     BASED ON RANGE 1-15
FE61 20EE                BRA    INIT1     WHEN RANGE = 0, RESET RANGE

               *

FE63 DE02      CHKUP     LDX    LOWER     CHECK FROM LOWER TO UPPER
FE65 8655                LDAA   #$55
FE67 9805                EORA   PASS
FE69 D606      CHKUP1    LDAB   PATT
FE6B 43                  COMA
FE6C A100      CHKUP2    CMPA   X
FE6E 2661                BNE    ERROR
FE70 43                  COMA
FE71 A700                STAA   X
FE73 43                  COMA
FE74 08                  INX
FE75 9C00                CPX    UPPER
FE77 2708                BEQ    CHKDN
FE79 5D                  TSTB
FE7A 27F0                BEQ    CHKUP2
FE7C 5A                  DECB
FE7D 26ED                BNE    CHKUP2
FE7F 20E8                BRA    CHKUP1

               *

FE81 43        CHKDN     COMA             CHECK FROM UPPER TO LOWER
FE82 09        CHKDN1    DEX
FE83 A100                CMPA   X
FE85 264A                BNE    ERROR
FE87 43                  COMA
FE88 A700                STAA   X
FE8A 43                  COMA
FE8B 9C02                CPX    LOWER
FE8D 279A                BEQ    PTEST1
FE8F 5D                  TSTB
FE90 27F0                BEQ    CHKDN1
FE92 5C                  INCB
FE93 D106                CMPB   PATT
FE95 23EB                BLS    CHKDN1
FE97 43                  COMA
FE98 C601                LDAB   #1
FE9A 20E6                BRA    CHKDN1

FE9C 8D26      DONE      BSR    PRINTSTR
FE9E 20564552            FCC    / VERIFIE/
FEA6 C4                  FCB    'D+$80
FEA7 20FE                BRA    *

               * PUTHEX OUTPUTS ASCII OF HEX IN ACCA

```

```
FEA9 36         111: PUTHEX  PSHA           SAVE A COPY
FEAA 44         112:         LSRA
FEAB 44         113:         LSRA
FEAC 44         114:         LSRA
FEAD 44         115:         LSRA
FEAE 8D01       116:         BSR    PUTHEX1
FEB0 32         117:         PULA
FEB1 840F       118: PUTHEX1 ANDA   #$F
FEB3 8B90       119:         ADDA   #$90    CONVERT TO ASCII
FEB5 19         120:         DAA
FEB6 8940       121:         ADCA   #$40
FEB8 19         122:         DAA
FEB9 37         123: PUTHEX2 PSHB           SAVE ACCUMULATOR
FEBA BDFC06     124:         JSR    PUTCV
FEBD 33         125:         PULB           RESTORE ACCUMULATOR
FEBE 39         126:         RTS
                127:
                128: * PRINT STRING SUBROUTINE.  PRST IS TOP OF STRING LOOP, NOT ENTRY POINT.
                129: * PRINTS STRING POINTED TO BY RETURN ADDRESS ON TOP OF STACK.  STRING
                130: * TERMINATED BY BIT 7 SET ON ON LAST CHAR.
                131:
FEBF 8DF8       132: PRST    BSR    PUTHEX2 OUTPUT DATA IN ACCA
FEC1 30         133:         TSX            INDEX INTO STACK
FEC2 6C01       134:         INC    1,X     ADVANCE RETURN ADDRESS POINTER
                135: ;       BNE    PRINTSTR        ** ** NOTE.  THIS BRANCH WILL ALWAYS BE
                136: ;       INC    0,X     IN BOOT ROM BECAUSE OF PLACMENT OF CALLS.
                137:
                138: * ENTRY POINT OF SUBROUTINE.
                139:
FEC4 30         140: PRINTSTR       TSX             ;NEEDED AT ENTRY TO SUBR
FEC5 EE00       141:         LDX    0,X     POINT TO DATA TO OUTPUT
FEC7 A600       142:         LDAA   0,X     GET DATA BYTE IN ACCA FOR OUTPUT
FEC9 2AF4       143:         BPL    PRST    B/ NOT AT END OF DATA STRING
FECB 8DEC       144:         BSR    PUTHEX2 OUTPUT LAST BYTE
FECD 30         145:         TSX            ADVANCE RETURN ADDRESS
FECE 6C01       146:         INC    1,X
FED0 39         147:         RTS
                148:
                149:
                150: * ERROR, REPORT GOOD, BAD, AND ADDRESS
                151:
FED1 E600       152: ERROR   LDAB   0,X     GET BAD BYTE
FED3 36         153:         PSHA           SAVE GOOD BYTE
FED4 DF02       154:         STX    LOWER   SAVE ADDRESS
FED6 8DEC       155:         BSR    PRINTSTR        PRINT HEADER
FED8 20455252   156:         FCC    / ERR @/
FEDE A0         157:         FCB    $20+$80
FEDF 9602       158:         LDAA   LOWER
FEE1 8DC6       159:         BSR    PUTHEX
FEE3 9603       160:         LDAA   LOWER+1
FEE5 8DC2       161:         BSR    PUTHEX  PRINT ADDRESS
FEE7 8DDB       162:         BSR    PRINTSTR
FEE9 20474F4F   163:         FCC    / GOOD/
FEEE BD         164:         FCB    '=+$80
FEEF 32         165:         PULA           GET GOOD BYTE
```

```
FEF0 8DB7                BSR     PUTHEX
FEF2 8DD0                BSR     PRINTSTR
FEF4 20424144            FCC     / BAD/
FEF8 BD                  FCB     '=+$80
FEF9 17                  TBA             GET BAD BYTE
FEFA 8DAD                BSR     PUTHEX
FEFC 20FE                BRA     *
```

```
               *       ROM I/O AREA

 FF00                  ORG     $FF00   START OF I/O AREA

               *
               * MINI-FLOPPY DISK CONTROLLER
               *
 FF80                  ORG     $FF80   DISK CONTROLLER ADDRESS
FF80 0002              RMB     2       PIA STATUS REGISTERS
FF82 0002              RMB     2       PIA DATA REGISTERS
FF84 0001              RMB     1       FD 1793 COMMAND/STATUS REGISTER
FF85 0001              RMB     1       FD 1793 CURRENT TRACK REGISTER
FF86 0001              RMB     1       FD 1793 NEXT SECTOR REGISTER
FF87 0001              RMB     1       FD 1793 NEXT TRACK REGISTER
               *
               * HYTYPE PARALLEL PRINTER CONTROLLER
               *
 FF90                  ORG     $FF90   HYTYPE PIA ADDRESS
FF90 0002              RMB     2       STATUS REGISTERS
FF92 0002              RMB     2       DATA REGISTERS
               *
               * PERSCI FLOPPY DISK CONTROLLER
               *
 FFA0                  ORG     $FFA0   DISK CONTROLLER ADDRESS
FFA0 0002              RMB     2       PIA STATUS REGISTERS
FFA2 0002              RMB     2       PIA DATA REGISTERS
FFA4 0001              RMB     1       FD 1771 COMMAND/STATUS REGISTER
FFA5 0001              RMB     1       FD 1771 CURRENT TRACK REGISTER
FFA6 0001              RMB     1       FD 1771 NEXT SECTOR REGISTER
FFA7 0001              RMB     1       FD 1771 NEXT TRACK REGISTER
               *
               * WAVE MATE VIDEO KEYBOARD (SCREEN AT $C000)
               *
 FFB0                  ORG     $FFB0   VIDEO PIA ADDRESS
FFB0 0002              RMB     2       STATUS REGISTERS
FFB2 0002              RMB     2       DATA REGISTERS
```

```
             *
             * SERIAL PORTS
             *
  FFC0               ORG    $FFC0   SERIAL ACIA ADDRESS AREA
FFC0 0002            RMB    2       SERIAL PORT 0
FFC2 0002            RMB    2       UNASSIGNED
FFC4 0002            RMB    2       SERIAL PORT 1
FFC6 0002            RMB    2       UNASSIGNED
FFC8 0002            RMB    2       SERIAL PORT 2
FFCA 0002            RMB    2       UNASSIGNED
FFCC 0002            RMB    2       SERIAL PORT 3
FFCE 0002            RMB    2       UNASSIGNED
FFD0 0002            RMB    2       SERIAL PORT 4 (OVERLAY CASSETTE I/O)
FFD2 0002            RMB    2       UNASSIGNED
FFD4 0002            RMB    2       SERIAL PORT 5
FFD6 0002            RMB    2       UNASSIGNED
FFD8 0002            RMB    2       SERIAL PORT 6
FFDA 0002            RMB    2       UNASSIGNED
FFDC 0002            RMB    2       SERIAL PORT 7
FFDE 0002            RMB    2       UNASSIGNED
             *
             * CASSETTE PORT
             *
  FFD0               ORG    $FFD0   CASSETTE ADDRESS
FFD0 0002            RMB    2       CASSETTE ACIA
             *
             * EPROM PROGRAMMER PORT (2704-2708 EPROMS)
             *
  FFD4               ORG    $FFD4
FFD4 0001            RMB    1       DATA A
FFD5 0001            RMB    1       STATUS A
FFD6 0001            RMB    1       DATA B
FFD7 0001            RMB    1       STATUS B
             *
             * HARDWARE BREAKPOINT REGISTERS
             *
  FFD8               ORG    $FFD8   BREAKPOINT PIA ADDRESS
FFD8 0002            RMB    2       STATUS REGISTERS
FFDA 0002            RMB    2       DATA REGISTERS


             *       ROM INTERRUPT VECTORS
             *
  FFF8               ORG    $FFF8
FFF8 FC78            FDB    LEVEL0
FFFA FD21            FDB    SWIV
FFFC FC7F            FDB    LEVEL7
FFFE FC94            FDB    HARDRESET
  0000               END
```

```
Symbols Sorted by Name:

 ABS/****          BOOT/FD25         BOOTV/FC00        CHKDN/FE81        CHKDN1/FE82       CHKUP/FE63        CHKUP1/FE69       CHKUP2/FE6C
 CLR/FCA9          CMD/FD85          COMDATA/FD41      COUNT/0004       *CURCHR/0082      *CURH/0080        *CURL/0081        *CURSTAT/0098
 DISKACKDONE/FDAB                    DONE/FE9C         ERROR/FED1        ESCC/FC42         ESCCE/FC4C       *ESCCV/FC0F        FDCMDX/FD92
 FDWAIT/FD87      *FINT2/FC8E        FPITIM/FC8B       GETCV/FC09       *GETFV/FC18        GETS1/FDC7        GETS2/FDD5        GETSA/FDCA
 HARDRESET/FC94    IDBNMI/FC3D       INFOFF/FDFF       INFON/FDFC        INI5IN/FD74       INI8IN/FD61       INICV/FC03        INIFV/FC12
 INIHWD/FD95       INILV/FC27        INIS1/FE11        INIS2/FDB1        INISA/FDB4        INIT/FE4B         INIT1/FE51        INIT2/FE54
 INITIALIZE/FD61                     INTDISV/FC33     *INTENV/FC36      *IOFFFV/FC24      *IONFV/FC21        ISMINI/FDAD       LEVEL0/FC7B
 LEVEL7/FC7F      *LINSIZ/0040       LOOP/FCAF         LOWER/0002        NOINT/FC39        NOTENCRYPT/FCB7                     NULS1/0002
 NULS2/0002       *OOFFFV/FC1E      *OOFFLV/FC30      *OONFV/FC1B       *OONLV/FC2D        OUTFOFF/FE05      OUTFON/FE02       PASS/0005
 PATT/0006         PRINTSTR/FEC4     PRST/FEBF         PTEST1/FE29       PUTCV/FC06        PUTFV/FC15        PUTHEX/FEA9       PUTHEX1/FEB1
 PUTHEX2/FEB9     *PUTLV/FC2A        PUTS1/FDE6        PUTS1E/FDFB       PUTS2/FE0A        PUTSA/FDEB        PUTSB/FDEB        PUTSN/FDF5
 READSECT/FD50     RESET/FCD7        RESET9/FC9F       RETRY/FD2B        RETURN/FD94       RUNNINGENCRYPTED/00FF              *SCREEN/C000
*SCRSIZ/0B00       SETBOOT/FCF4      SETSECT/FD55      SKP2/008C         SOFTRESET/FD04    SWIV/FD21         SYSCCI/00D5      *SYSCLK/00E1
 SYSINIT/FC4D      SYSINT0/00FD     *SYSINT1/00FA     *SYSINT2/00F7     *SYSINT3/00F4     *SYSINT4/00F1     *SYSINT5/00EE     *SYSINT6/00EB
 SYSINT7/00E8      SYSLOOP/FCDE     *SYSMSK/00E0       SYSPG/00FE        SYSPWR/00DE       SYSRST/00D8       SYSSWI/00DB      *SYSVARS/00D3
*TABTBL/0083       TESTRAM/FE16      TOPAGE/FC00       TOPLOOP/FCBC     *TRANSP/0099       TSTBOOT/FD46      TSTCV/FC0C       *TSTMEM/FC3A
 TSTS1/FDBD       *TSTS10/FDDA       TSTSA/FDC0        TSTSAO/FDDD       UPPER/0000
```

```
Symbols Sorted by Value:

 ABS/****          UPPER/0000        LOWER/0002        NULS1/0002        NULS2/0002        COUNT/0004        PASS/0005         PATT/0006
*LINSIZ/0040      *CURH/0080        *CURL/0081        *CURCHR/0082      *TABTBL/0083       SKP2/008C        *CURSTAT/0098     *TRANSP/0099
*SYSVARS/00D3      SYSCCI/00D5       SYSRST/00D8       SYSSWI/00DB       SYSPWR/00DE      *SYSMSK/00E0      *SYSCLK/00E1       SYSINT7/00E8
*SYSINT6/00EB     *SYSINT5/00EE     *SYSINT4/00F1     *SYSINT3/00F4     *SYSINT2/00F7     *SYSINT1/00FA      SYSINT0/00FD      SYSPG/00FE
 RUNNINGENCRYPTED/00FF              *SCRSIZ/0800      *SCREEN/C000       BOOTV/FC00        TOPAGE/FC00       INICV/FC03        PUTCV/FC06
 GETCV/FC09        TSTCV/FC0C       *ESCCV/FC0F        INIFV/FC12        PUTFV/FC15       *GETFV/FC18       *OONFV/FC1B       *OOFFFV/FC1E
*IONFV/FC21       *IOFFFV/FC24       INILV/FC27       *PUTLV/FC2A       *OONLV/FC2D       *OOFFLV/FC30       INTDISV/FC33     *INTENV/FC36
 NOINT/FC39       *TSTMEM/FC3A       IDBNMI/FC3D       ESCC/FC42         ESCCE/FC4C        SYSINIT/FC4D      LEVEL0/FC78       LEVEL7/FC7F
 FPITIM/FC8B      *FINT2/FC8E        HARDRESET/FC94    RESET9/FC9F       CLR/FCA9          LOOP/FCAF         NOTENCRYPT/FCB7
 TOPLOOP/FCBC      RESET/FCD7        SYSLOOP/FCDE      SETBOOT/FCF4      SOFTRESET/FD04    SWIV/FD21         BOOT/FD25         RETRY/FD2B
 COMDATA/FD41      TSTBOOT/FD46      READSECT/FD50     SETSECT/FD55      INI8IN/FD61       INITIALIZE/FD61                     INI5IN/FD74
 CMD/FD85          FDWAIT/FD87       FDCMDX/FD92       RETURN/FD94       INIHWD/FD95       DISKACKDONE/FDAB                    ISMINI/FDAD
 INIS2/FDB1        INISA/FDB4        TSTS1/FDBD        TSTSA/FDC0        GETS1/FDC7        GETSA/FDCA        GETS2/FDD5       *TSTS10/FDDA
 TSTSA0/FDDD       PUTS1/FDE6        PUTSA/FDEB        PUTSB/FDEB        PUTSN/FDF5        PUTS1E/FDFB       INFON/FDFC        INFOFF/FDFF
 OUTFON/FE02       OUTFOFF/FE05      PUTS2/FE0A        INIS1/FE11        TESTRAM/FE16      PTEST1/FE29       INIT/FE48         INIT1/FE51
 INIT2/FE54        CHKUP/FE63        CHKUP1/FE69       CHKUP2/FE6C       CHKDN/FE81        CHKDN1/FE82       DONE/FE9C         PUTHEX/FEA9
 PUTHEX1/FEB1      PUTHEX2/FEB9      PRST/FEBF         PRINTSTR/FEC4     ERROR/FED1

 129 Symbols.
```

```
Error Lines:

JUPITERROM26.ASM
   1-19      2-2

 ### 2 Errors.
```